4 A1 creating the resource data-containing file, which is not complied, using a markup language to identify the parameter; using a graphical control locator for identifying the resource data-containing file; parsing text in the resource data-containing file; walking the parsed text to provide the parameter to the graphical interface; and carrying out the instructions in the markup language to implement the graphical interface in accordance with the parameter.

SB A2 33. (Amended) A method for enhancing user control over the look and feel of an application program executable on a computer with the help of an operating system, the application program using at least one external resource file and computer executable instructions, the method comprising the steps of: creating the resource file using a markup language; providing a resource-loader routine for retrieving information from the resource file without compiling the resource file by making a call in the computer executable instructions to the resource-loader to obtain information from the resource file; generating the computer executable instructions independent of the resource file; and executing the instructions in the markup language to provide the requested information to the resource-loader.

34. (Amended) A system for developing an application program for execution on a computer with the help of an operating system, the application program having at least one resource file and computer executable instructions, the system comprising: the resource file containing information stored in accordance with a markup language; an executable file corresponding to the application including computer executable instructions for making a call on the resource loader; a resource-loader routine in the operating system for retrieving information

from the resource file without compiling the resource file and in response to a call made by the executable file while executing; and an interpreter for executing the instructions in the markup language to provide the requested information to the resource-loader.

35. (Amended) A resource loader in an operating system for accessing information 2 in a resource file, which is not complied, during execution of an application program and providing the information to the application program wherein the resource loader program comprises: an addressing mechanism for communicating a request from an application program for specified information from at least one resource file; a markup language handling functionality to retrieve data stored using a markup language wherein the data in the resource file is modifiable by a user while the application is executing and wherein the data corresponds to the information requested by the application program from the resource loader; and a second addressing mechanism for directly or indirectly communicating the retrieved data to the application program from the resource loader.

36. (Amended) A computer readable medium having computer executable instructions for carrying out the steps of a method for developing an application program, the application program having at least one graphical interface, the graphical interface having at least one parameter specified in a resource data-containing file, the method comprising the steps of: 5 creating the resource data-containing file, which is not complied, using a markup language to identify the parameter; using a graphical control locator for identifying the resource data-containing file; parsing text in the resource data-containing file; walking the parsed text to